5 Post Flight Data Processing

Upon completion of each flight, files are transferred off the N3R data acquisition system to a ground-based PC for post-flight processing. These data are put through a series of post-flight processing programs with each performing a specific task. All programs are in written in GNU C and are executed on a PC under a Linux operating system. The following is a brief outline of the processing steps. Raw GPS data from the aircraft and ground station are combined to produce a set of differentially-corrected GPS aircraft positions and velocities. Next, the *in situ* and remote sensor data are merged with the differentially corrected GPS data and converted into a network common data format (netCDF). Quality control programs are implemented on these data to remove spikes and outliers and to perform a series of other data quality checks. Finally, the calibrations are applied, low- and high-rate data are merged, winds are computed, and dynamic heating corrections are applied to the air temperature. During each step, time stamps are written in the output file header. These stamps describe when the data were processed and the version of the processing programs. This information is carried through to the final product. Additionally, the marker files are manually edited to reflect special notes regarding the flight and designate special legs for analysis purposes. Figure 18 contains a flow chart illustrating the steps for data post-processing. The rectangles in the figure represent specific data files while the ovals represent individual programs. Table 2 is a summary of the various acquired and processed data files.

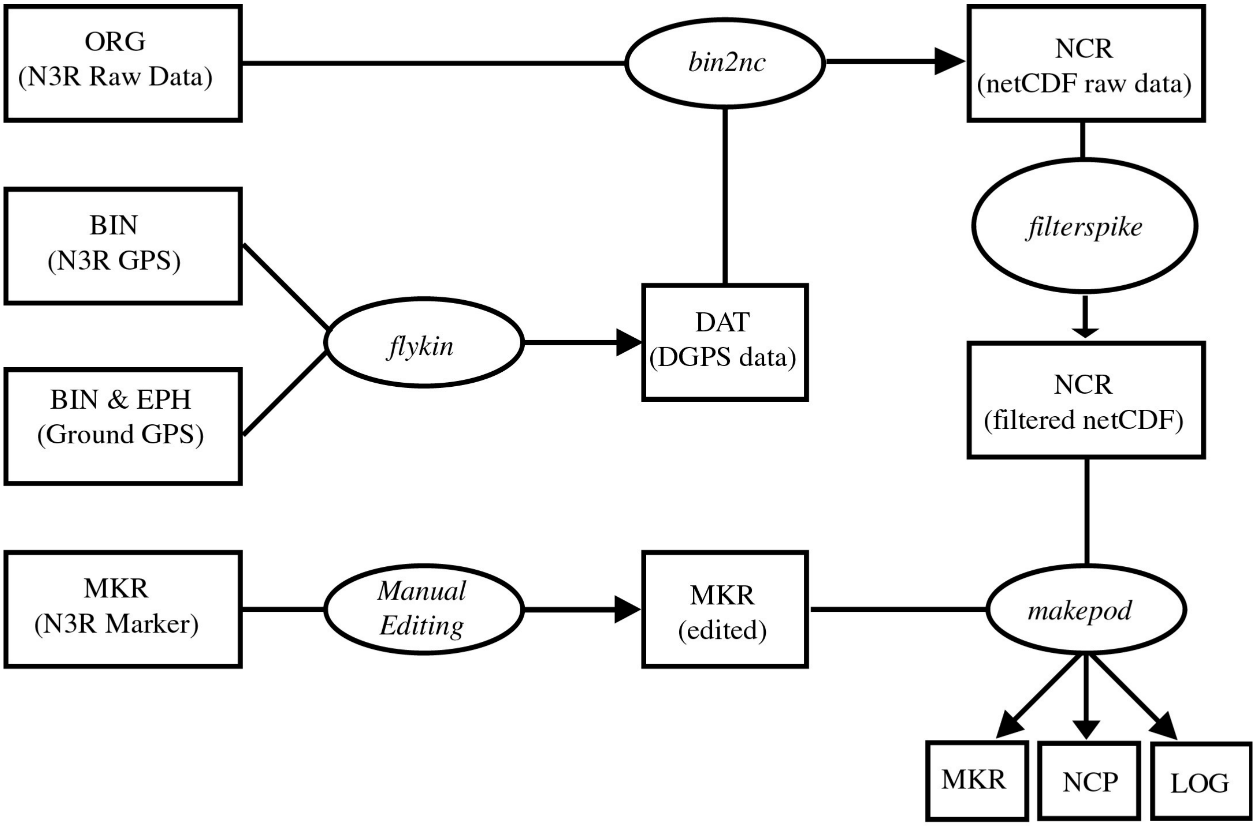

Fig. 18. Flowchart summarizing N3R data post-processing steps.

5.1 Differential GPS Corrections

The initial step in post-flight processing calculates positions and velocities using differential GPS techniques. Although the raw GPS data from N3R contains positions and velocities, accuracies are coarse (even after the elimination of selective availability) and are of little scientific use. The raw GPS data used in the corrections algorithm contain satellite navigation information and, for each epoch, signal phase, pseudo-range, and satellite number. Based on these data collected from two separate locations, one of which is precisely known (the ground-station), it is possible to calculate very accurate position and speed of the remote observing station (N3R). For this purpose, an algorithm (*flykin*) developed by the Department of Geomatics Engineering at the University of Calgary is used (Cannon et al. 1993). The algorithm, written in C, uses advanced filtering techniques and phase information to resolve ambiguities resulting from the determination of position based solely on pseudo-range. The source code for this software package was modified by ARL for our application on N3R. The algorithm creates an ASCII file of precise GPS information with the same root name and a DAT extension.

5.2 NetCDF Conversion

The next step in the data post-processing converts the files from their native format into netCDF (Rew et al. 1997). This program (*bin2nc*) executes a simple file conversion of the ORG data file. It also merges output from the differentially corrected GPS data, replacing the uncorrected GPS data stored in the original aircraft files. During the merging process, flags are set for missing GPS data. Also, because data are stored either as short- or long-word integers, a scale factor and offset must be computed. These values are assigned to variable attributes in the netCDF file. This binary file has the same root name with a NCR (netCDF "raw") extension.

5.3 Quality Control

The next program applies quality control checks on the raw NCR data file and applies corrections as necessary. A despike routine (*filterspike*) checks the data for regions where values fall outside of a prescribed range or are equal to some predefined "fill" value. In these regions, data are linearly interpolated based on the last and next "good" value. After completing the interpolation, flags are set for those regions where data were interpolated. Flags are set to mark these records.

5.4 Final Processing

The final step in the post processing algorithms is to carry through the actual computation of winds, apply recovery factors based on aircraft velocity, and to merge data from slow- and fast-response sensors. A program (*makepod*) calculates winds based on the raw pressure measurements from the BAT probe in combination with data from the GPS. High- and low-frequency measurements of position, velocity, temperature, and water vapor are blended by first performing Fourier transforms followed by passing the data through appropriate high- and low-pass filters, merging the sets, and finally calculating the inverse transform (Eckman et al. 1999). The final output file has the same root name and is given the extension NCP (netCDF "processed"). In addition, a LOG file is created by *makepod*. This ASCII file contains calibration information and statistics (e.g., mean, standard deviation, skewness, kurtosis, minimum, maximum) for all variables.

Wind calculations use a method similar to the National Center for Atmospheric Research (NCAR) method described by Leise and Masters (1991). Modifications to the NCAR method result from the unique configuration of the BAT probe in that a reference pressure (not a true static pressure) is provided by the pneumatic average of the four reference pressure ports. Consequently, the dynamic pressure must be corrected not only for variations in attack angle and sideslip, but also for deviations of reference pressure from the static pressure. The computation of accurate wind measurements requires the empirical determination of several BAT probe calibration constants. Of particular interest are the angle offsets, roll, pitch, heading, angle of attack offset, and dynamic pressure adjustment (Table 3). The values of these constants are determined by minimizing the variance of the three individual wind components for various flight legs. These legs include straight and level flight, dynamic and steady-state pitch and yaw maneuvers, reverse-heading maneuvers, and wind circles and boxes. Theoretically determined constants, based on theory of potential flow over a sphere are also included in the wind calculations. These include attack and sideslip calibration constants and the aircraft upwash factor.

Table 2. Summary of data files.

File	Description	Format	Output from:
mmddhhmn.ORG	raw aircraft data	binary	N3R data acquisition system
mmddhhmn.BIN	Ashtech GPS data	binary	N3R data acquisition system
mmddhhmn.MKR	marker data	ASCII	N3R data acquisition system
ammdd-0x.BIN	Ashtech GPS data	binary	ground station
ammdd-0x.EPH	Ashtech GPS data	binary	ground station
mmddhhmn.DAT	DGPS data	ASCII	*flykin*
mmddhhmn.NCR	netCDF raw data	binary	*bin2nc*, *filterspike*
mmddhhmn.NCP	netCDF processed data	binary	*makepod*
mmddhhmn.LOG	calibrations, data statistics	ASCII	*makepod*

mm	month
dd	day
hh	hour
mn	minute
a	“Ashtech ground station”
x	= 1, 2, 3, etc. (if multiple flights are conducted during the same day)

Table 3. Calibration constants and switches used in *makepod*.

Constant	Value	Description
LicorSw	0	switch for LICOR 6262 (0 = off, 1 = on)
LaserSW	1	switch for laser altimeters (0 = off, 1 = on)
RadarSw	1	switch for Ka-band radar scatterometer (0 = off, 1 = on)
EG&GSw	1	switch for EG&G dew point sensor (0 = off, 1 = on)
FlightQ	9	minimum Px for which winds are computed
Pad	50	maximum elements to pad arrays for FFT edge effect reduction
R_T	0.82	temperature recovery factor
$"_0$	-0.0663225	angle of attack of DSP at zero lift
$K_{"}$	0.24	pitch calibration constant
$K_{\$}$	0.19	yaw calibration constant
K_{up}	0.101	upwash factor
$,_q$	1.0175	dynamic pressure correction
$,_p$	-1.43	pitch offset for relative velocity
$,_r$	0.0	roll offset for relative velocity
$,_h$	-0.5	heading offset for relative velocity
Apass	0.75	highest frequency passed by low-pass filter (attitude) highest frequency stopped by high-pass filter (attitude)
Astop	1.333	lowest frequency stopped by low-pass filter (attitude) lowest frequency passed by high-pass filter (attitude)
Vpass	0.25	highest frequency passed by low-pass filter (velocity) highest frequency stopped by high-pass filter (velocity)
Vstop	0.444	lowest frequency stopped by low-pass filter (velocity) lowest frequency passed by high-pass filter (velocity)
Hpass	0.0075	highest frequency passed by low-pass filter (hor. position) highest frequency stopped by high-pass filter (hor. position)
Hstop	0.01333	lowest frequency stopped by low-pass filter (hor. position) lowest frequency passed by high-pass filter (hor. position)
Wpass	0.0375	highest frequency passed by low-pass filter (vert. position) highest frequency stopped by high-pass filter (vert. position)
Wstop	0.0666	lowest frequency stopped by low-pass filter (vert. position) lowest frequency passed by high-pass filter (vert. position)